

```

#include <string.h>
#include <sys/socket.h>
#include <netdb.h>
#include <stdlib.h>
#include <stdio.h>
#include <netinet/in.h>
#include <sys/types.h>
#include <netinet/in.h>
#define BUF_SIZE 256

int main(int argc, char *argv[])
{
    struct sockaddr_in server, client;
    char str[50] = "hi";
    char mail_f[50], mail_to[50], msg[20], c;
    int t = 0;
    socklen_t l = sizeof(server);
    if (argc != 3)
        printf("Input format not correct");
    int sockfd = socket(AF_INET, SOCK_DGRAM, 0);
    if (sockfd == -1)
        printf("Error in socket()");
    server.sin_family = AF_INET;
    server.sin_addr.s_addr = INADDR_ANY;
    server.sin_port = htons(atoi(argv[2]));
    client.sin_family = AF_INET;
    client.sin_addr.s_addr = INADDR_ANY;
    client.sin_port = htons(atoi(argv[2]));
    printf("Sending hi to server");
    sleep(10);
    if (sendto(sockfd, str, sizeof(str), 0, (struct sockaddr *)&server,
sizeof(server)) < 0)
        printf("Error in sento");
    if (recvfrom(sockfd, str, sizeof(str), 0, (struct sockaddr *)&server, &l) <
0)
        printf("Error in recv");
    printf("\ngreeting msg is %s", str);
    if (strncmp(str, "220", 3))
        printf("\nConn not established \n code 220 expected");
    printf("\nSending HELO");
    strcpy(str, "HELO 127.0.0.1");
    if (sendto(sockfd, str, sizeof(str), 0, (struct sockaddr *)&server,
sizeof(server)) < 0)
        printf("Error in sendto");
    sleep(3);
    printf("\nReceiving from server");
    if ((recvfrom(sockfd, str, sizeof(str), 0, (struct sockaddr *)&server, &l))
< 0)
        printf("Error in recv");
    if (strncmp(str, "250", 3))
        printf("\nOk not received from server");
    printf("\nServer has send %s", str);
    printf("\nEnter FROM address\n");
    scanf("%s", mail_f);
    strcpy(str, "MAIL FROM");
    strcat(str, mail_f);
    if (sendto(sockfd, str, sizeof(str), 0, (struct sockaddr *)&server,
sizeof(server)) < 0)
        printf("Error in sendto");
    sleep(3);
    if ((recvfrom(sockfd, str, sizeof(str), 0, (struct sockaddr *)&server, &l))
< 0)
        printf("Error in recv");
    if (strncmp(str, "250", 3))

```

```

        printf("\nOk not received from server");
        printf("%s", str);
        printf("\nEnter TO address\n");
        scanf("%s", mail_to);
        strcpy(str, "RCPT TO");
        strcat(str, mail_to);
        if (sendto(sockfd, str, sizeof(str), 0, (struct sockaddr *)&server,
sizeof(server)) < 0)
            printf("Error in sendto");
        sleep(3);
        if ((recvfrom(sockfd, str, sizeof(str), 0, (struct sockaddr *)&server, &l))
< 0)
            printf("Error in recv");
        if (strcmp(str, "250", 3))
            printf("\nOk not received from server");
        printf("%s", str);
        printf("\nSending DATA to server");
        strcpy(str, "DATA");
        if (sendto(sockfd, str, sizeof(str), 0, (struct sockaddr *)&server,
sizeof(server)) < 0)
            printf("Error in sendto");
        sleep(3);
        if ((recvfrom(sockfd, str, sizeof(str), 0, (struct sockaddr *)&server, &l))
< 0)
            printf("Error in recv");
        if (strcmp(str, "354", 3))
            printf("\nOk not received from server");
        printf("%s", str);
        printf("\nEnter mail body");
        while (1)
        {
            c = getchar();
            if (c == '$')
            {
                msg[t] = '\0';
                break;
            }
            if (c == '\0')
                continue;
            msg[t++] = c;
        }
        if (sendto(sockfd, msg, sizeof(msg), 0, (struct sockaddr *)&server,
sizeof(server)) < 0)
            printf("Error in sendto");
        sleep(3);
        printf("\nSending QUIT to server");
        strcpy(str, "QUIT");
        if (sendto(sockfd, str, sizeof(str), 0, (struct sockaddr *)&server,
sizeof(server)) < 0)
            printf("Error in sendto");
        if ((recvfrom(sockfd, str, sizeof(str), 0, (struct sockaddr *)&server, &l))
< 0)
            printf("Error in recv");
        if (strcmp(str, "221", 3))
            printf("\nOk not received from server");
        printf("\nServer has send GOODBYE.....Closing conn\n");
        printf("\n Bye");
        close(sockfd);
        return 0;
}

```